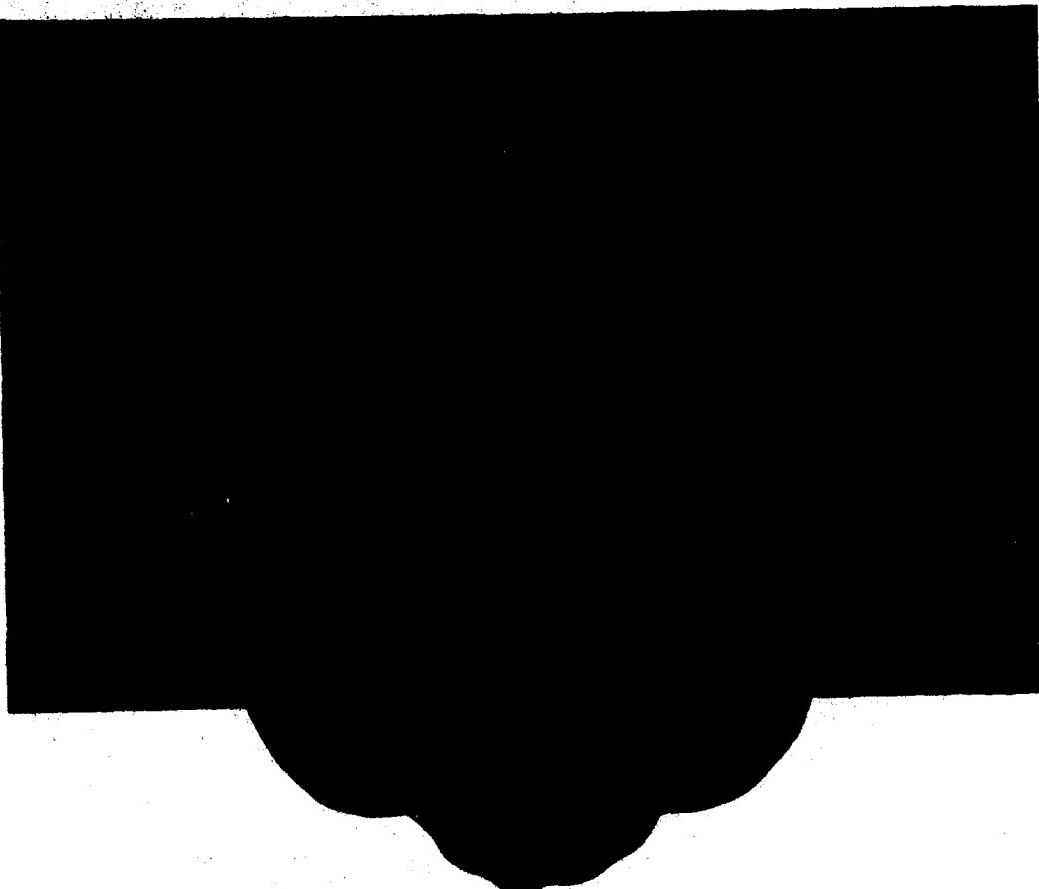


AD-A115 804 NORTH CAROLINA UNIV AT CHAPEL HILL INST OF STATISTICS F/G 12/1
OPTIMIZATION ALGORITHMS ON RANDOM GRAPHS, (U)
APR 82 D G KELLY, D J WELSH N00014-76-C-0550
UNCLASSIFIED Mimeo SER-1389 NL

Jord
411A
5804

END
DATE
FIRMED
107-82
DTIC

AD A115804



OPTIMIZATION ALGORITHMS ON RANDOM GRAPHS

by

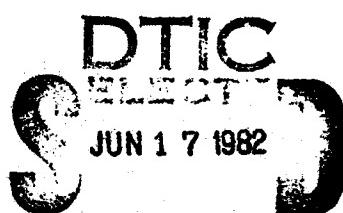
D. G. Kelly

and

D. J. A. Welsh

Institute of Statistics/Mimeo Series #1389

April 1982



FILE COPY

DEPARTMENT OF STATISTICS
Chapel Hill, North Carolina

This document has been approved
for public release and sale; its
classification is unclassified.

82 06 17 012

Optimization algorithms on random graphs.

D. G. Kelly* and D. J. A. Welsh

I. Introduction.

Let G be a graph on vertices $0, 1, \dots, n-1$. We consider two 'algorithms' which start at an arbitrary vertex, move at each step to a lower-numbered neighboring vertex, and stop when they reach a sink, i.e. a vertex with no lower-numbered neighbor:

The greedy algorithm moves from a given vertex to the lowest-numbered neighbor;

The random algorithm moves from a given vertex to a vertex chosen at random from the lower-numbered neighbors.

We are interested in the behavior of these two algorithms on random graphs. We use the model of random graphs in which each of the $\binom{n}{2}$ edges is present or absent with probabilities p and $q=1-p$ respectively, independently of the other edges. Such a random graph will be called a $\text{Bern}(n,p)$ graph. We will consider the following random variables:

S_n , the number of sinks in a $\text{Bern}(n,p)$ graph;

G_k , the number of steps from vertex k to a sink using the greedy algorithm on a $\text{Bern}(n,p)$ graph;

R_k , the number of steps from vertex k to a sink using the random algorithm on a $\text{Bern}(n,p)$ graph.

These considerations are motivated by probabilistic analysis of linear programming, which attempts to reconcile the widely-observed efficiency of the

* Research supported by ONR Grant No. N00014-76-C-0550.

Abstract

We consider random graphs on vertices $0, 1, 2, \dots, n$ in which each edge, independent of the others, is present with probability p and absent with probability $q=1-p$. On such a graph we consider two different random walks starting with vertex n , moving at each step from a vertex to a lower-numbered neighbor, and stopping when they reach either vertex 0 or a sink, a vertex with no lower-numbered neighbor. These walks are simplified attempts to reproduce probabilistically the behavior of the simplex algorithm on linear programs.

We derive some simple asymptotic results on the distribution of the number of sinks in a random graph, and also on the distributions of the numbers of steps needed by each of the two random walks.

A

simplex algorithm with the existence of problems on which it is exponentially inefficient.

If a linear program in d variables with m constraints is generated at random from one of a wide class of distributions [DKT], its feasible region is with probability one a d -dimensional polyhedron that is simple (each vertex has exactly d neighbors), in which no two vertices have the same objective value. Thus it is possible to number the vertices $0, 1, \dots, n-1$ so that 0 is the optimal vertex and so that from any vertex, any simplex pivot moves to a lower-numbered neighboring vertex.

In another approach to probabilistic analysis [K], one considers a fixed linear program and regards the simplex algorithm as a random walk on the graph of its feasible region which moves always in the direction of improvement in objective value. This is obviously the motivation for the random algorithm described above.

Our graph-algorithm models fail to match the above situations in two respects. First, the random graphs generated are not regular as are the graphs of simple polyhedra. This is a relatively minor defect, however; by proper choice of p and n one can adjust the average degree so that the actual vertex degrees will be close to d ; and one feels that it is not the regularity, but rather the fact that the vertex degrees are bounded below, that contributes to the efficiency of pivoting procedures. Second, and more important, the random graphs generated will in general have many sinks, while the polyhedron graphs have only one. Nevertheless we feel that an investigation of the behavior of such algorithms on graphs may shed some light, at least on the combinatorial reasons of the simplex algorithm's efficiency (as opposed to reasons related to convexity).

Another difference between random graphs and random linear programs is that the former have a fixed number of vertices (and random vertex degree) while the latter have fixed vertex degree but a random number of vertices. The distribution of the number n of vertices is not known, although one has [Klee]

$$(m-d)(d-1) + 2 \leq n \leq \left(\frac{m-a}{m-d}\right) + \left(\frac{m-b}{m-d}\right)$$

(where a and b are, respectively, the greatest integers in $(d+1)/2$ and $(d+2)/2$); and for a certain class of distributions on polyhedra with m facets in d dimensions [KT]

$$E(n) \sim (cd)^{d/2} m \text{ as } m \rightarrow \infty \text{ for large } d$$

where c is a constant independent of d and m .

In a $\text{Bern}(n,p)$ graph the average vertex degree is approximately np , and so we might be interested in the case of large n and $p = p_n$ satisfying

$$np \sim d \text{ and } n \sim d^{d/2},$$

i.e.

$$n \sim (np)^{np/2}.$$

II. Sinks in random acyclic directed graphs.

A. Let G be a $\text{Bern}(n,p)$ graph with vertices $0, 1, \dots, n-1$, and regard any edge as directed from the higher- to the lower-numbered vertex. A vertex of outdegree zero is a sink; we let $S = S_{n,p}$ be the number of sinks in G . (S is at least 1 because vertex 0 must be a sink.) Define random variables X_j ($j=0, \dots, n-1$) by

$$X_j = \begin{cases} 1 & \text{if } j \text{ is a sink,} \\ 0 & \text{if not.} \end{cases}$$

So $S = \sum_{j=0}^{n-1} X_j$, the X_j are independent, and

$$E[X_j] = \Pr(X_j=1) = q^j. \quad (1)$$

It follows immediately that the characteristic function and the probability generating function of S are, respectively,

$$\varphi_S(t) = E(e^{itS}) = \prod_{j=0}^{n-1} (1 - (1-e^{it})q^j),$$

$$\pi_S(u) = E(u^S) = \prod_{j=0}^{n-1} (1 - (1-u)q^j).$$

We also have

$$ES = 1 + q + q^2 + \dots + q^{n-1} = (1-q^n)/p$$

and, since $VX_j = q^j(1-q^j)$ and the X_j are independent,

$$VS = q(1-q^{n-1})(1-q^n)(1-q^{2n-1})$$

and

$$VS/(ES)^2 = pq(1-q^{n-1})(1+q)^{-1}(1-q^n)^{-1}.$$

Notice also that S is asymptotically normal for large n .

B. Now we consider $p = p_n$, $q = q_n = 1-p_n$ as functions of n and look at the behavior of $S = S_n$ as n increases for various orders of magnitude of p_n .

1. If p_n tends to a constant p ($0 < p < 1$), then ES_n approaches $1/p$ and VS_n approaches $q/p(1+q)$ (where $q=1-p$), and the distribution of S_n tends to the normal distribution with these parameters. (In general, if p_n is bounded away

from 0, then $ES_n = 1/p_n$ and $VS_n = q_n/p_n(1+q_n)$ both approach zero.)

2. If $p_n \rightarrow 0$ as $n \rightarrow \infty$, then $VS_n/(ES_n)^2 \rightarrow 0$ also, and it follows that

$S_n/ES_n \rightarrow 1$ in probability. We consider the asymptotic behavior of ES_n for a few growth rates of p_n .

a. If $p = (\log n)/n$, then

$$\begin{aligned} ES_n &\sim (n/\log n)(1 - (1-(\log n)/n)^n) \\ &\sim (n/\log n)(1-n^{-1}) \\ &\sim n/\log n. \end{aligned}$$

b. If $np_n \rightarrow \alpha$ ($0 < \alpha < \infty$), then

$$\begin{aligned} ES_n &\sim p_n^{-1}(1 - (1-p_n)^n) \\ &\sim p_n^{-1}(1-e^{-\alpha}) \\ &\sim (1-e^{-\alpha})n/\alpha. \end{aligned}$$

c. If $np_n \rightarrow 0$ but $n^{(k+1)/k} \rightarrow \alpha$ ($0 < \alpha < \infty$), then since

$$ES_n = n - \binom{n}{2}p + \binom{n}{3}p^2 - \dots$$

and since the partial sums in this expression alternately over- and

underestimate ES_n , eventually ES_n is between

$$n - \binom{n}{2}p + - \dots \pm \binom{n}{k}p^{k-1}$$

and

$$n - \binom{n}{2}p + - \dots \pm \binom{n}{k}p^{k-1} + \binom{n}{k+1}p^k;$$

and the last term on the right approaches $\bar{\alpha}^k$ as $n \rightarrow \infty$.

III. Number of steps needed by greedy algorithm.

Again fix n , p , and k ($k = 0, 1, \dots, n-1$); G_k is the number of steps from k to a sink using the greedy algorithm on a $\text{Bern}(n, p)$ graph.

Vertex k is a sink, and hence $G_k = 0$, with probability q^k ; otherwise the first step is to vertex j with probability pq^j ($j = 0, 1, \dots, k-1$). Consequently the characteristic function of G_k is

$$\varphi_k(t) = E(e^{itG_k}) = q^k + \sum_{j=1}^{k-1} pq^j E(e^{it(1+G_j)}),$$

i.e.

$$\varphi_k(t) = q^k + e^{it} \sum_{j=1}^{k-1} pq^j \varphi_j(t), \quad (2)$$

Replacing k by $k+1$ in (2) and subtracting (2) from the result yields

$$\varphi_{k+1}(t) - \varphi_k(t) = q^{k+1} - q^k + e^{it} pq^k \varphi_k(t),$$

whence

$$\varphi_{k+1}(t) = (1 + pe^{it} q^k) \varphi_k(t) - pq^k \quad (k > 0),$$

$$\varphi_0(t) = 1.$$

Similarly, for the probability generating function,

$$\pi_{k+1}(u) = (1 + pq^k u) \pi_k(u) - pq^k, \quad (3)$$

A. To get an expression for $\varphi_k(t)$ write $p_k(t) = e^{it} \varphi_k(t) - 1$. Then one gets

$$p_{k+1}(t) = p_k(t)(1 + pq^k e^{it}).$$

Since $p_0(t) = e^{it} - 1$ we have

$$p_k(t) = (e^{it} - 1) \prod_{j=0}^{k-1} (1 + pq^j e^{it})$$

and hence

$$\varphi_k(t) = e^{-it} + (1 - e^{-it}) \prod_{j=0}^{k-1} (1 + pq^j e^{it}).$$

B. Let $\mu_k = EG_k = \pi'_k(1)$. Differentiating (3) and setting $u = 1$ gives

$$\mu_{k+1} = pq^k + (1 + pq^k) \mu_k.$$

Writing $\alpha_k = \mu_k + 1$ we get

$$\alpha_{k+1} = (1 + pq^k) \alpha_k,$$

so that

$$\alpha_k = \prod_{j=0}^{k-1} (1+pq^j)$$

and

$$\mu_k = \prod_{j=0}^{k-1} (1+pq^j) - 1.$$

Notice that regardless of the values of n and p , because of the arithmetic mean

- geometric mean inequality we have

$$\begin{aligned} 1 + \mu_k &< (1 + p(1+q+q^2+\dots+q^{k-1})/k)^k \\ &= (1 + (1-q^k)/k)^k \\ &< (1 + 1/k)^k. \end{aligned}$$

It follows that

$$\mu_k < e - 1 \text{ for all } k = 0, 1, \dots$$

IV. Number of steps needed by random algorithm.

Let R_k be the number of steps from k to a sink using the random algorithm on a $\text{Bern}(n,p)$ graph. With probability q^k vertex k is a sink and $R_k = 0$. With probability $1-q^k$ vertex k is not a sink; in this case by symmetry all vertices of lower index than k have the same chance of being visited next. Accordingly the characteristic function of R_k is

$$\psi_k(t) = E(e^{itR_k}) = q^k + k^{-1}(1-q^k) \sum_{j=0}^{k-1} E(e^{it(R_j+1)}),$$

i.e.

$$\psi_k(t) = q^k + k^{-1}(1-q^k)e^{it} \sum_{j=0}^{k-1} \psi_j(t).$$

Similarly, the probability generating function is

$$\lambda_k(u) = q^k + k^{-1}(1-q^k)u \sum_{j=0}^{k-1} \lambda_j(u),$$

Hence, if $\mu_k = ER_k = \lambda'_k(1)$, then $\mu_0 = 0$ and

$$\mu_k = (1-q^k)(1+k^{-1} \sum_{j=0}^{k-1} \mu_j).$$

A. To get a quick upper bound on μ_k (a better bound will be obtained in B. below) we note that if α_k is defined by the recurrence

$$\alpha_k = 1 + k^{-1} \sum_{j=0}^{k-1} \alpha_j \quad (\text{with } \alpha_0 = 0),$$

then it is easy to show that $\mu_k \leq \alpha_k$ and also that

$$(k+1)\alpha_{k+1} - k\alpha_k = 1 + \alpha_k,$$

whence

$$\alpha_k = \alpha_{k-1} + 1/k.$$

Thus we have

$$\mu_k \leq \sum_{j=1}^k 1/j \sim \log k.$$

B. For more precise estimates of μ_k we write

$$c_k = 1 - q^k$$

so that

$$\mu_k = c_k(1 + k^{-1} \sum_{j=0}^{k-1} \mu_j). \tag{4}$$

Putting $k+1$ in place of k in (4), subtracting (4) from the result, and

simplifying, we find

$$\beta_{k+1} = r_{k+1}\beta_k + s_{k+1},$$

where

$$r_j = (c_j/c_{j-1})(1-j^{-1}q^{j-1})$$

and

$$s_j = j^{-1}c_j,$$

It follows easily that

$$\beta_k = s_k + r_k s_{k-1} + r_k r_{k-1} s_{k-2} + \dots + r_k r_{k-1} \dots r_2 s_1. \quad (5)$$

Moreover:

$$s_k = k^{-1}(1-q^k),$$

$$r_k s_{k-1} = (k-1)^{-1}(1-q^k)(1-k^{-1}q^{k-1}),$$

and in general

$$r_k \dots r_{j+1} s_j = j^{-1}(1-q^k)(1-k^{-1}q^{k-1})(1-(k-1)^{-1}q^{k-2}) \dots (1-(j+1)^{-1}q^j). \quad (6)$$

It follows easily that

$$\beta_k \leq (1-q^k) \sum_{j=1}^k 1/j \sim (1-q^k) \log k.$$

A number of different lower bounds on β_k can be obtained from (5) and (6).

Among them are:

$$\beta_k \geq 1-q^k,$$

$$\beta_k \geq (1-q^k)(1-q/2)^k \sum_{j=1}^k 1/j,$$

and

$$\beta_k \geq (1-q^k)(2 + q^{-1} \log(1-q)) \sum_{j=1}^k 1/j.$$

V. Open problems

(There is no guarantee that any of the problems posed here are difficult, or that their solutions will constitute major progress. There is also no guarantee that any of them are well-posed or tractable.)

A. For the greedy algorithm, find a closed-form expression for the characteristic function $\varphi_k(t)$ of the number G_k of steps from k to a sink. Also find the variance of G_k , and get asymptotic values for the mean and variance. Show that the expected value μ_k is an increasing function of k , and also an increasing function of p .

B. For the random algorithm, attempt to find a closed-form expression for the characteristic function $\psi_k(t)$ of the number R_k of steps from k to a sink. Find a closed-form expression for the expected value β_k , and for the variance as well. Find asymptotic expressions or bounds on these quantities. Show that the expected value β_k is increasing as a function of k and of p . Find an asymptotic lower bound on β_k ($\log k$ would be nice).

C. Look into controlling the number of sinks in the random graphs under consideration, either by adjusting p as a function of n (probably this can be done only with unreasonable values of p) or by changing the model somehow. Look into threshold-function results for various behaviors; for example, is there a threshold growth rate on $p = p_n$ for boundedness of the sequence (β_k) ?

VI. References.

[DKT] Dunham, J., Kelly, D., and Tolle, J. Some experimental results concerning the expected number of pivots for solving randomly generated linear programs. Technical Report no. 77 - 16, UNC Curriculum in Operations Research and Systems Analysis, Chapel Hill, NC (1977).

[K] Kelly, D. Some results on random linear programs. Operations Research Verfahren (1981), 351 - 353.

[KT] Kelly, D. and Tolle, J. Expected number of vertices of a random convex polyhedron. SIAM J. Alg. and Disc. Methods 2 (1981), 441-451.

[Klee] Klee, V. Polytope pairs and their relationship to linear programming. Acta Mathematica 133 (1974), 1 - 25.

